دراسة عامة للمقارنة بين خوارزميتي التشفير DES و TDES

ميمونة حميد الحداد
Shrm_4fra@yahoo.com

العراق - جامعة الكوفة/ كلية التربية للبنات / قسم الحاسبات

الخلاصة :

التشفير هو عملية الحفاظ على سرية المعلومات باستخدام برامج لها القدرة على تحويل وترجمة تلك المعلومات إلى رموز بحيث إذا ما تم الوصول إليها من قبل أشخاص غير مخول لهم بذلك لا يستطيعون فهم أي شيء لأن ما يظهر لهم هو خليط من الرموز والأرقام والحروف الغير مفهومة.

لذلك تعبر كلمة " تشفير" عن تحويل أو " بعثرة" البيانات إلى هيئة غير قابلة للفهم لإرسالها عبر وسط ناقل معين إلى جهة محددة بحيث لا يمكن لأي جهة غير الجهة المقصودة تفسير هذه البيانات المبهمة واستخلاص البيانات المفهومة منها.

ولذلك فأن الرغبة في حماية المعلومات والبيانات من السرقة وإساءة الاستخدام أدت إلى ظهور تقنيات التشفير . وان تقنيات التشفير هي الطريقة الكفوءة لتوفير أمنية المعلومات والبيانات لذلك السبب بذل الكثير من الخبراء والعلماء سعيهم لتطوير هذه التقنيات من خلال زيادة درجة تعقيدها ، وقد ظهرت أنواع عديدة لهذه التقنيات حسب الحاجة لها حيث أن أخفاء المعلومات المفيدة في صورة غير مفهومة هو الحلم الذي حققه التشفير . ومن خوارزميات التشفير المشهورة خوارزمية تشفير البيانات القياسي ((DES)Data Encryption Standard) وهذه الخوارزمية تعتمد في عملها على إدخال كتلة من البيانات مكونة من 64 بت وتشفيرها بواسطة مفتاح يتكون أيضا من 64 بت وDES تعامل هذه الكتلة من البيانات ذات 64 بت في ما يسمى بالشفرة الكتلوية المدورة أي انه DES تستخدم 16 دورة يتم خلالها تشفير كتلة البيانات المحددة.

لذلك دأب الخبراء على زيادة أمنية خوارزمية DES من خلال زيادة درجة تعقيدها، وذلك بواسطة زيادة عدد المفاتيح المستخدمة في التشفير حيث استخدمت ثلاثة مفاتيح مختلفة في كل دورة بدلاً من المفتاح الواحد، ولذا سميت خوارزمية DES المضاعفة TDES.

ويتناول هذا البحث عملية المقارنة بين خوارزميتي التشفير DES و TDES وذلك من حيث العمل والمضمون وسرعة التنفيذ وعدد المفاتيح المستخدمة في عملية التشفير.

General study on the comparison between encoding algorithm DES and TDES

Abstract

Encoding is the process of saving information security by using programs having the capability to transform and translate these information in to symbols in such away that if unrelated people know them, they cannot understand anything because whatever appears to them will be mixture of anonymous symbols, numbers and letters therefore, encoding means the transformation and disarrangement of data in to anonymous form in order to send them by a certain milieu to an accurate direction in away that no one can illustrate these data except the intended direction. So the techniques of encode result from the desire to secure information and data from theft and bad use. Encode is an excellent way to save information and data, so experts make their high efforts to develop encoding by in creasing its complexity. Many types of encoding techniques appear as a result of need, and we can consider that hiding information is a dream becomes true by encode. Data Encryption Standard (DES) is one of the most famous algorithms, this algorithm depends on entering amass of data consists of 64 bits and encoded it by using a key consists of 64 bits in what we called round mass code. This means that DES uses sixteen circulations. Thus, the intended data will be encoding. So the experts persevered at increasing DES algorithm security by increasing its complexity by three different keys at every round instead of one. Thus DES named TDES.

This research concerned with the comparison between encode algorithm DES and TDES regarding work, essence, performance speed and the number of keys using in encoding.

1- المقدمة:

في سنة 1972 قام المكتب القومي للتقيس في الولايات المتحدة الأمريكية " National Bureau of Standard"التابع للقسم التجاري ، بالبدء بتصميم نظام قياسي لحماية المعلومات التي تعالج بواسطة الحاسبة الالكترونية من كافة التهديدات و المخاطر المتعمدة و غير المتعمدة ، وقد قام هذا المكتب بوضع المتطلبات الأساسية والإطار العام لأمنية الحاسبات الالكترونية وطلب من الجهات المعنية بالأمر بتصميم نظام يلبي المتطلبات أعلاه لاعتماده وتطبيقه في حماية بيانات الحاسبة وقد التمس هذا المكتب من وكالة الأمن الوطني (NSA) بتقييم الأنظمة التي ستصل من الجهات الأخرى أو القيام بتصميم النظام في حالة عدم قيام جهة أخرى بتصميمه.

بالمقابل كانت شركة (IBM) قد الفت مجموعة عمل بقيادة د.هورست فيستل في نهاية الستينات للبحث في مجال تشفير البيانات وقد أوجدت نظام يسمى (LUOIFFER) وفي بداية السبعينات تولى الدكتور تجمان قيادة فريق العمل هذا وقد نتج عن هذه الجهود وضع نظام التشفير القياسي للبيانات. ولقد قامت شركة (IBM) بتقديم هذا النظام إلى وكالة الأمن الوطني (NSA) .

وقد أطلق عليه اسم نظام تشفير البيانات القياسي (DES) Data Encryption Standard))

ويستند هذا النظام إلى خوارزمية لوسيفر (Lucifer algorithm) التي تستخدم مفتاح تشفير بطول 56 بت ، وتشترط أن يكون لكل من المرسل والمستقبل المفتاح السري ذاته . وقد استخدمت الحكومة هذا النظام عام 1976 واعتمدته في الكثير من المجالات ومنها في البنوك لتشغيل آلات الصرف الآلي (ATM).

ان الـDES هو نظام تشفير معقد ولا خطي وقادر على العمليات ذات السرع العالية عندما يتم تنفيذه بالمكونات المادية التي تسمح به.

أن خوارزمية الـ DES يحول 64 بت من النص الصريح الى 64 بت من النص المشفر تحت تأثير مفتاح طوله 64 بت . ان المقطع المطلوب تشفيره يتعرض لترتيبات أولية ، وبعد ذلك الى حسابات معقدة تعتمد على المفتاح ، وأخيرا إلى ترتيبات معكوسة بالنسبة للترتيبات الأولية.

أن سلسلة الحسابات هي ربط متوالي من الـ16دورة ، كل دورة تستخدم 48 بت من المفتاح في سلسلة تحسب بواسطة قائمة المفاتيح والتي تؤمن عملية مزج بتات المفتاح لكل دورة.

باستثناء هذا الفرق في المفاتيح الدوارة ، فان الـ16 دورة تكون متشابهة وكل دورة تستقبل مدخلاً 64 بت . والـ32 بت التي تمثل النصف الأيمن تفك بواسطة العامل الخطي E إلى 48 بت وتضاف النتيجة بالأساس الثنائي إلى مفتاح الدورة K .ان حاصل جمع الـ48 بت يقسم الى 8 مقاطع ذات 6 بتات وكل منها تدخل الى صناديق الـS الذي يعطي 4 بت كإخراج حيث ان الـ32بت الناتجة تضاف

حسب الأساس الثنائي الى النصف الأيسر.ومن ثم يتم تبادل المواقع لكلا النصفين و ينتج 64 بت تكون خارج الدورات.

أن الغرض من التبادل هو مزج بتات البيانات بحيث لايمكن استرجاعها ثانية من صناديق الـS التي هي عبارة عن جداول تعويضية لا خطية . وان هذه التقنية تقوي الخوارزمية و تجعلها مقاومة لهجوم محلل الشفرة.

2- مميزات DES:

1- كل عضو من مجموعة المستخدمين المخولين لاستخدام البيانات المشفرة في الحاسبة يجب إن يمتلك المفتاح الذي استخدم في تشفير البيانات وذلك لكي يستطيع ان يفك شفرة البيانات التي يقوم باستلامها.

2- خوارزمية التشفير تكون معروفة من قبل المستخدمين لهذا النظام، فيكون سري حيث فقط المستخدمين المخولين لهم العلم به.

3- سرية التشفير (أمنية التشفير) تعتمد على السرية التي يتم توفيرها للمفتاح المستخدم في عملية تشفير وحل الشفرة.

4- إن ألـDES يعتبر النظام الأول من ناحية نشر الخوارزمية الخاصة به. حيث إن محلل الشفرة يجب إن يحل عدد ضخم من معادلات الأنظمة اللاخطية لحل شفرة ألـ DES.

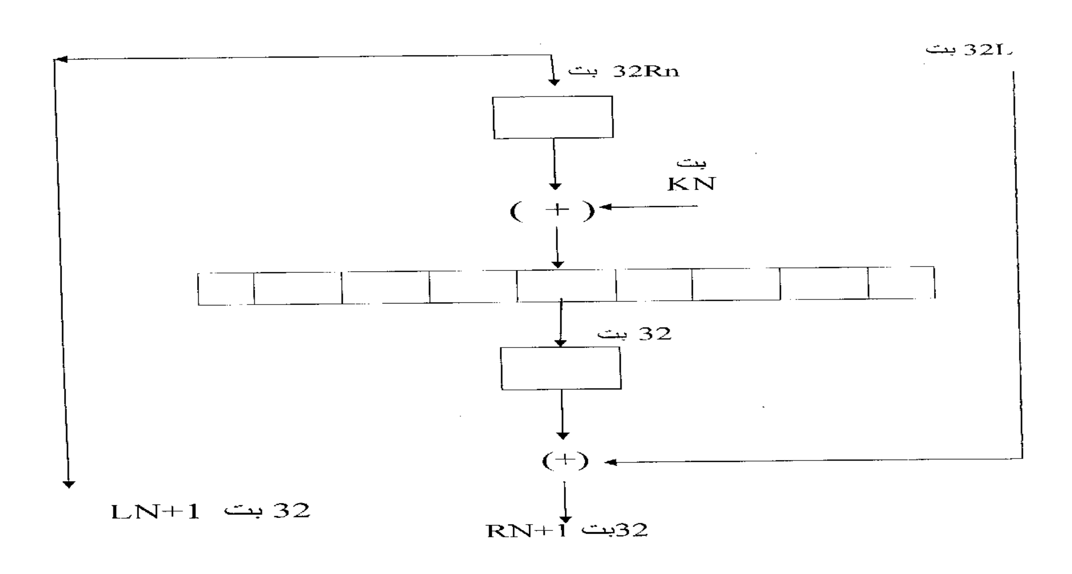
5- في أنظمة التشفير السابقة إذا كانت عملية التشفير معروفة فان تجربة كل المفاتيح يمكننا من حل النص المشفر ولكن مع ألـ DES كل الذي نعرفه هو كيفية عمل الخوارزمية وان مصممي الخوارزم يصرون على المفتاح الحالي ذا ألـ 64بت وهو بطول كافاً لجعل استخدام تقنيات التجربة والخطأ لكسر النص المشفر عملية غير مجدية.

3- هياكل البيانات المستخدمة:

نظام التشفير القياسي DES يستخدم ويحتاج هياكل البيانات آلاتية:

3-1 حزمة البيانات المعدة للتشفير:

البيانات المتكونة من 64 بت تقسم إلى جانب أيسر (L) ذات 32 بت وجانب ايمن (R) ذات 32بت ، وكما هو موضح في الشكل رقم (1).



شكل (1) تمثيل البيانات المعدة للتشفير

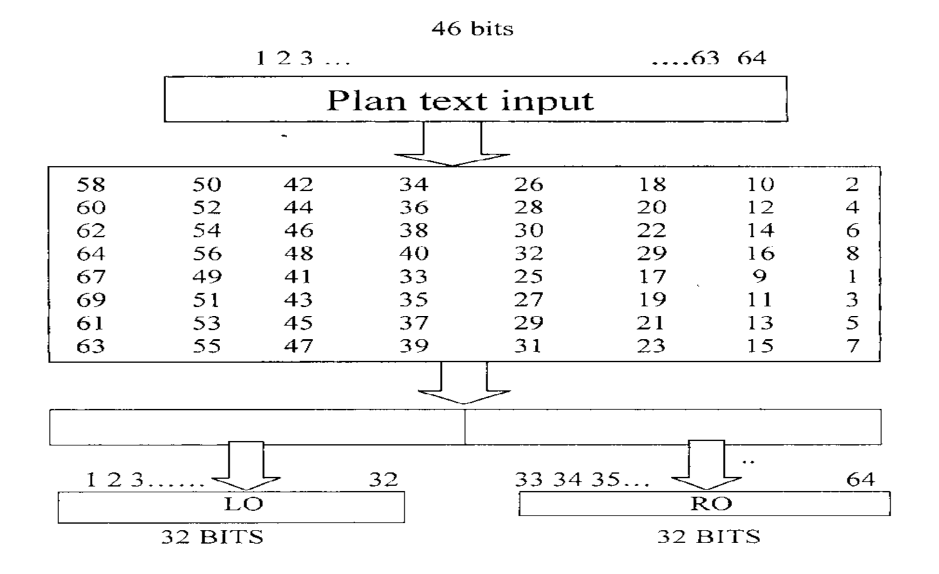
2-3 مفتاح التشفير K:

المتكون من الـ64 بت يشتق منه 16مفتاح فرعي طول كل منها 48 بت

K1,K2.......k16

3 -3 جدول الترتيب الأولي (IP) initial Permutation:

إن ألـ64 بت الداخلة تبدل بموجب موقع البت الموضح في جدول الـIP لتوليد مقطع بيانات بـ64 بت ، ومن ثم يقسم إلى جانب أيسر(32 بت) و ايمن(32 بت) . هذا الجدول يستخدم مرة واحدة لكل مقطع دخل .



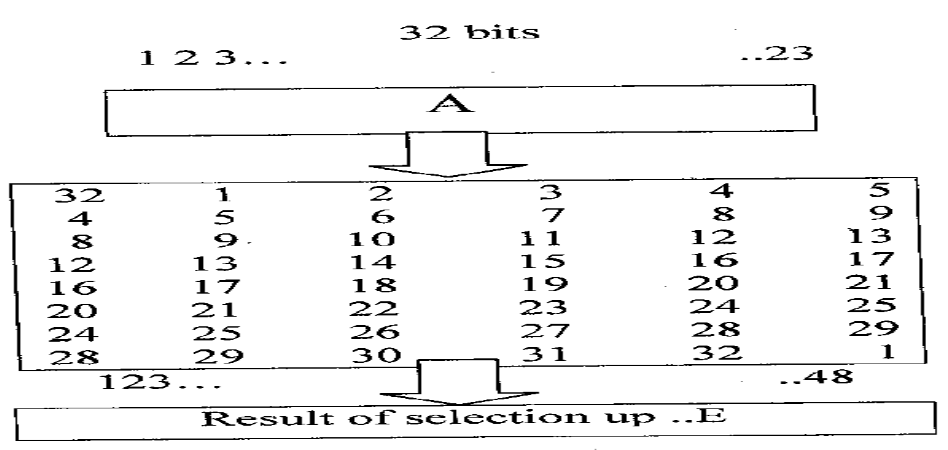
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	29	16	8
67	49	41	33	25	17	9	1
69	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

THE INITIAL PERMUTATION(IP)

جدول (1) للترتيب الاولي

3-4 جدول اختيار البت -E-Bit Selection table E:-

هذا الجدول يأخذ 32 بت كإخراج وذلك بمزج البتات تبعا للمواقع المؤشرة في الجدول رقم(2) ، الذي يُستخدم 16 مرة (ونلاحظ إن العديد من مواقع البتات في الجدول تكرر نفسها وهذا يجعل التوسع ممكنا).



32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

جدول (2) لاختيار البت- E

3-5 جدول اختيار الترتيب Permuted Choise-1 PC-1 :

هذا الجدول يبدل المفتاح الاصلي ذا 64 بت ليولد مفتاح ذا 56 بت يتكون من جزئين 28 لكل منها C0 على اليسار و D0 على اليمين و(كما في جدول (3)) والذي يستخدم مرة واحدة لكل مقطع للمدخلات

57	49	41	33	25	17	9	CO
1	58	50	42	34	26	18	
10	2	59	51	43	35	27	
19	11	3	60	52	44	36	
63	55	47	39	31	23	15	DO
7	62	45	46	38	30	22	
14	6	61	53	45	37	29	
21	13	5	28	20	12	4	

جدول (3) لاختيار الترتنيب PC-1

3-6 جدول الازاحة اليسرى Left Shift LS :

ان لكل دورة من 16 دورة للخوارزمية تستخدم مفتاح مختلف واعتماداً على رقم الدورة لذا فانه يتم تزحيف المفتاح ذا 56 بت اما بمقدار موقع واحد او موقعين الى اليسار ، (وكما موضح في الجدول (4)).

عدد الازاحات لليسار	رقم الدورة
1	16,9,2,1
2	10,8,7,6,5,4,3
2	11,12,13,14,15

جدول (4) لعدد الازاحات في كل دورة

3 -7 جدول اختيار الترتيب PERMUTED CHOICE-2 PC-2:

هو جدول الترتيب الذي يحول المفتاح ذا الـ 56 بت الناتج من جدول الـLS إلى مفتاح ذا 48 بت الذي سوف يضاف(جمع للأساس 2) إلى 48 بت المشتقة من جدول الـ E . هذا الجدولُ يستخدم 16 مرة (مرة واحدة لكل دورة). حسب جدول (5).

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	93	56	34	53
46	42	50	36	92	32

جدول (5) لاختيار البت PC-2 2

8- 3 صناديق التعويض SUBSTITUTION BOXES S:

جدول (6) يوضح صناديق التعويض التي هي عبارة عن ثمانية صناديق $S_1, S_2.......S_8$ كل صندوق أو جدول يأخذ 48 بت الناتجة من العملية $E+K_N$ كإدخال ذات 6 بت إلى كل صندوق وان الناتج هو مقطع من32 بت، وهذه الجداول الثمانية تُستخدم 16 مرة(مرة واحدة لكل دورة).

Columns / Rows	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S2	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	19
	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S3	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S4	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
S5	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S6	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S7	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S8	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	1
	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

جدول (6) لصناديق التعويض

3-9 جدول الترتيب P PERMUTATION:

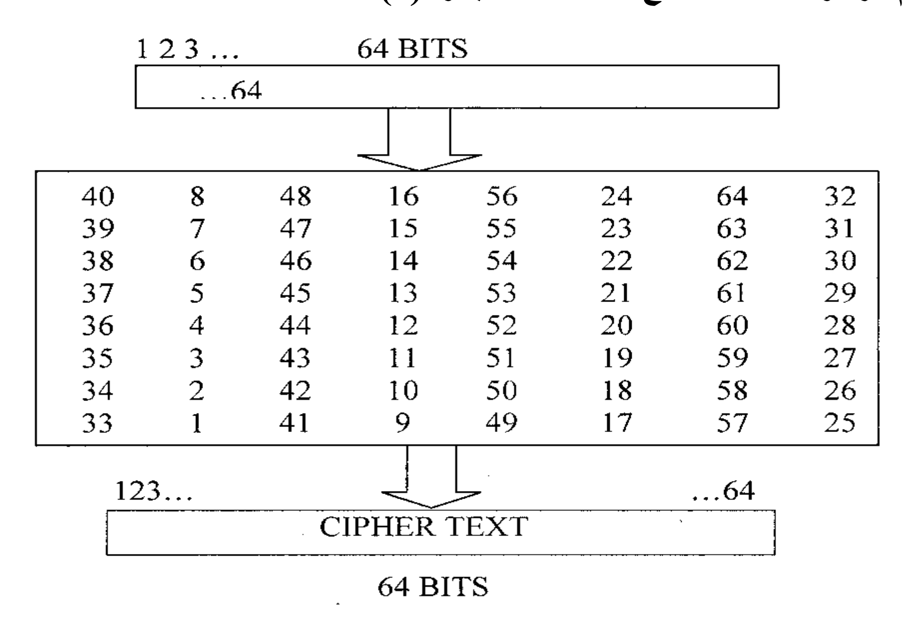
إن الإخراج من صناديق S والمكون من 32 بت ينقل إلى المواقع المثبتة في جدول(7).

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

جدول (7) للترتيب P

10- 3 جدول الترتيب الأولي المعكوس IP^{-1} :

إن 46 بت من الإخراج الناتج من 16 دورة تحول إلى مواقع البت الموضحة في جدول IP^4. هذا الجدول يستخدم مرة واحدة لكل مقطع دخل، لاحظ جدول(8).



40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

جدول (8) للترتيب المعكوس IP^{-1}

4- عملية التشفير:

إن الشكل (2) يوضح مخطط لعملية التشفير بعد أجراء الترتيب الأولي للبتات المكونة للكتلة المطلوب تشفيرها . حيث تقسم كتلة البيانات إلى قسمين كتلة L ذات 32 بت وتتبع بكتلة R ذات 32 بت ليعطي كتلة

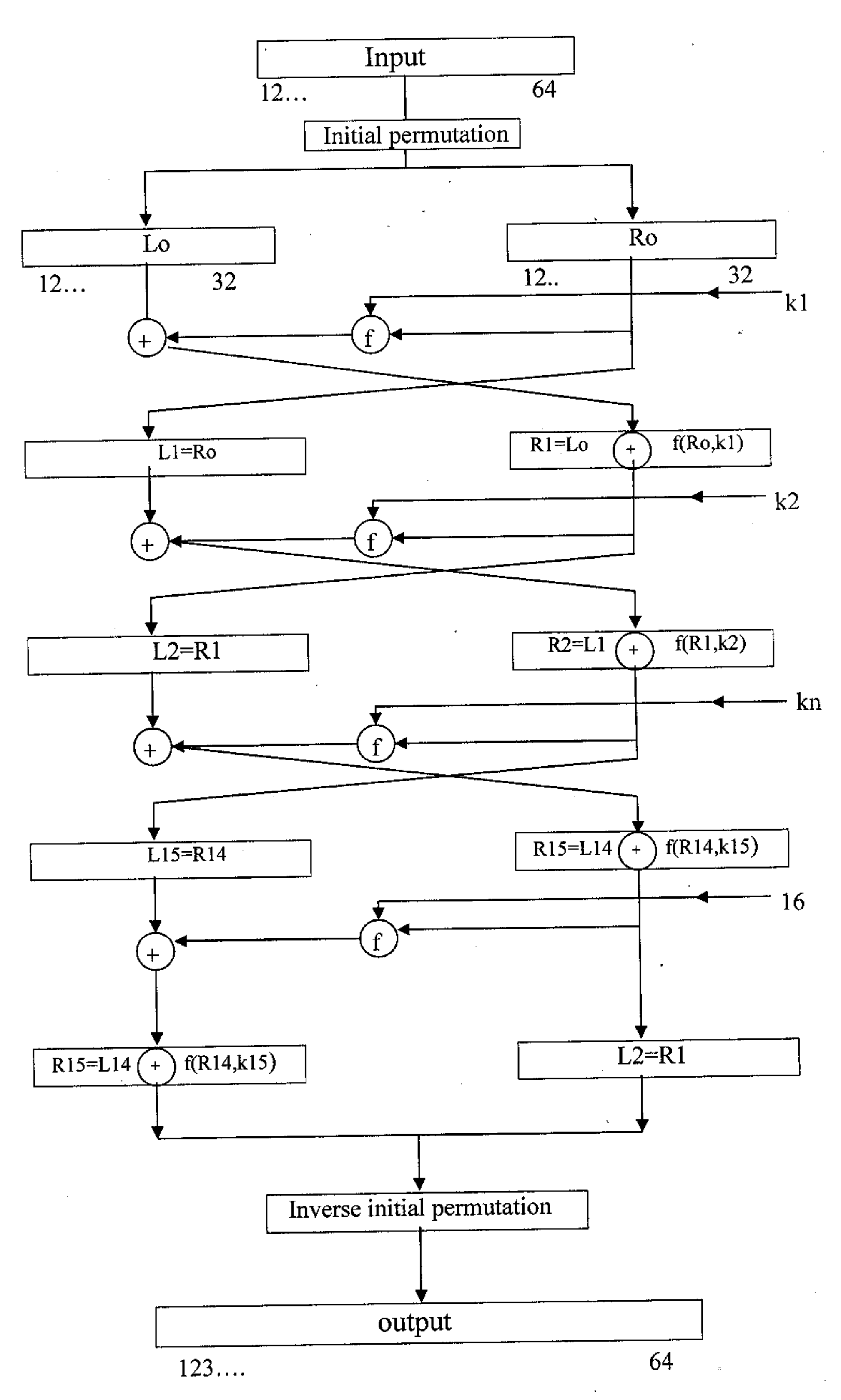
الإدخال ذات 64 بت والتي تسمى بـLR ، نفرض إن K هي كتلة تتألف من 48 بت اختيرت عشوائياً من المفتاح الرئيسي المتكون من 64 بت ليعطي ناتج الدورة الاولى LR

L=R R=L F (R,K)

حيث إن + هي عملية جمع للأساس، تكرر هذه العملية في 16 دورة يوجد مفتاح خاص مشتق من المفتاح الرئيسي وكما موضح في شكل (2) وبعد انتهاء ألـ16 دورة يتم عكس الترتيب الأولي بالاعتماد على جدول (IP^{-1}) جدول (8) وتكون عملية التشفير قد تمت.

$$L_N=R_{N-1}$$

$$R_N=L_N \oplus f(R_{N-1}, K_N)$$



شكل (2) يوضح عملية التشفير في DES

5- فك الشفرة:

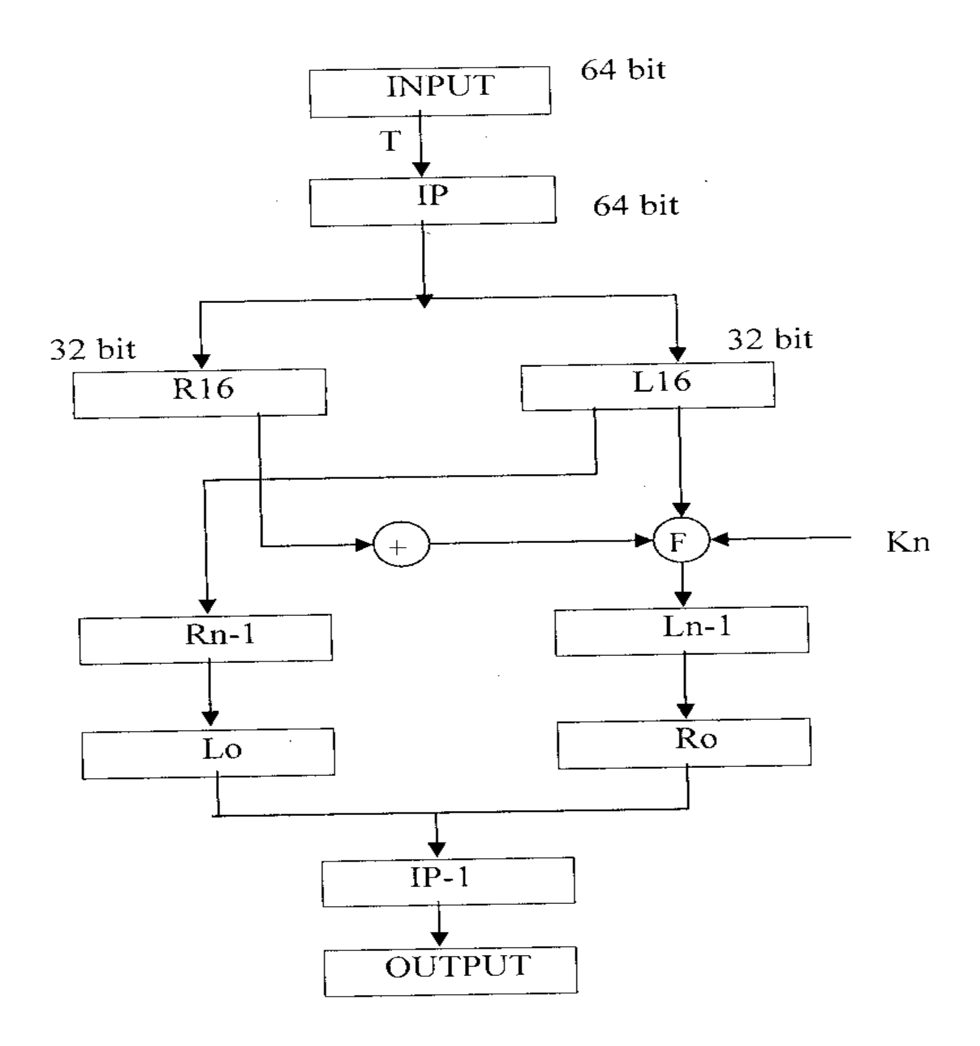
لفك الشفرة الناتجة من خوارزمية ألـDES فان كتلة البيانات المشفرة يجب ان تدخل أولاً إلى الخوارزمية وخلال عملية فك الشفرة فان نفس المفتاح (K_N) المستخدم خلال التشفير يجب إن يستخدم في التحليل وكلاً حسب دورته.

وتتم عملية فك الشفرة بأجراء الترتيب الأولي IP (جدول 1) كخطوة أولى ، ثم باستخدام نفس رموز عملية التشفير فان فك الشفرة يكون كالأتي:

$R_{N-1}=L_N$

$L_{N-1}=R_N \oplus f(L_N, K_N)$

حيث إنL_{16} وR_{16} هي كتلة الدخل المرتب لحساب تحليل الشفرة باستخدام K_{16} في الدورة الأولى وK_{15} في الدورة الثانية وهذا إلىK_1 الذي يستخدم في الدورة16. والشكل (3) يوضح عملية فك الشفرة.



شكل (3) يوضح عملية فك الشفرة

6- دالة التشفيرF:

شكل (4) يوضح عملية حساب دالة التشفير f(R,K) . إن الدالة E تأخذ كتلة البيانات المكونة من32 بت الطرف الأيمن كمدخلات وتنتج كتلة بيانات متكونة من 48 بت . وذلك بتقسيم كتلة البيانات إلى 8 كتل وكل كتلة متكونة من 6 بت حيث تكرر البيانات الأخيرة من كل كتلة وتكون هذة البتات هي البت الأولى والثانية ، وكما موضح في جدول (2).

إن الغاية من دالة ألـE هي جعل كتلة البيانات التي طولها 32بت تتوسع الى 48 وذلك لجعلها مساوية لطول المفتاح K الذي يتكون من 48 بت.

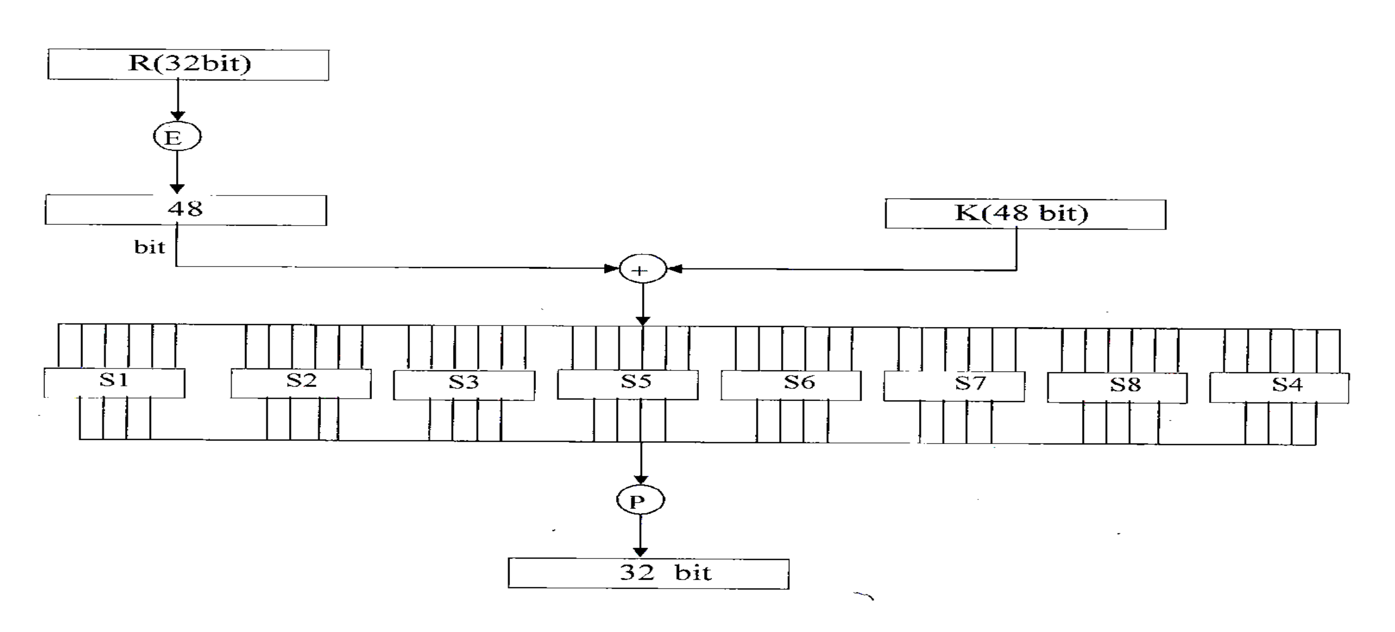
بعدها تطبق عملية (جمع للأساس 2) على كتلة البيانات المتولدة من الدالة E وكتلة المفتاح K ويكون ناتج هذة العملية هو 48 بت مقسمة إلى 8 كتل كل واحدة ذا 6 بت.

ألان يجب إن نحصل على 32 بت من ألـ48 بت من خلال 8 صناديق للتعويض جدول (6) $S_1, S_2....S_8$ ، حيث إن ناتج العملية السابقة المقسم إلى 8 كتل تدخل كل كتلة ذا 6 بت إلى صندوق ليعطي إخراج بـ4 بت وبذلك نحصل على 32 بت مجموع الكتل الخارجة من الصناديق الثمانية ويكون عمل صندوق التعويض كالأتي:

مدخلات كل صندوق 6 بت مثلا 100101

$B_1=1 \quad B_2=0 \quad B_3=0 \quad B_4=1 \quad B_5=0 \quad B_6=1$

حيث إن الأولى والأخيرة يمثل رقم السطر والبت الأخرى رقم العمود أي B_1 و B_6 يمثلان رقم السطر أي 11=3 (السطر الثالث) والبتات الأخرى أي B_2, B_3, B_4, B_5 تمثل رقم العمود أي 0010=2 (العمود الثاني).



شكل (4) يوضح عملية حساب دالة التشفير (R,K)

ولنفرض بان العمل على الصندوق S_1 فان القيمة الموجودة في السطر الثالث والعمود الثاني هي 8 وبعد تحويلها إلى النظام الثنائي نحصل على 1000 التي تتكون من 4 بت وبذلك حصلنا على 4 بت من 6 بت باستخدام صندوق التعويض ليكون مجموع الكتلة الناتجة 32 بت.وبعد ذلك يجري لهذة الكتلة المتكونة من 32 بت ترتيبا باستخدام جدول الترتيب P (جدول 7).

7- توليد المفتاح:

إن عملية توليد 16 مفتاح فرعي بطول 48 بت من كتلة بيانات تتكون من 64 بت يتم اختيارها عشوائياً توضح كالأتي(وكما في شكل(5)):

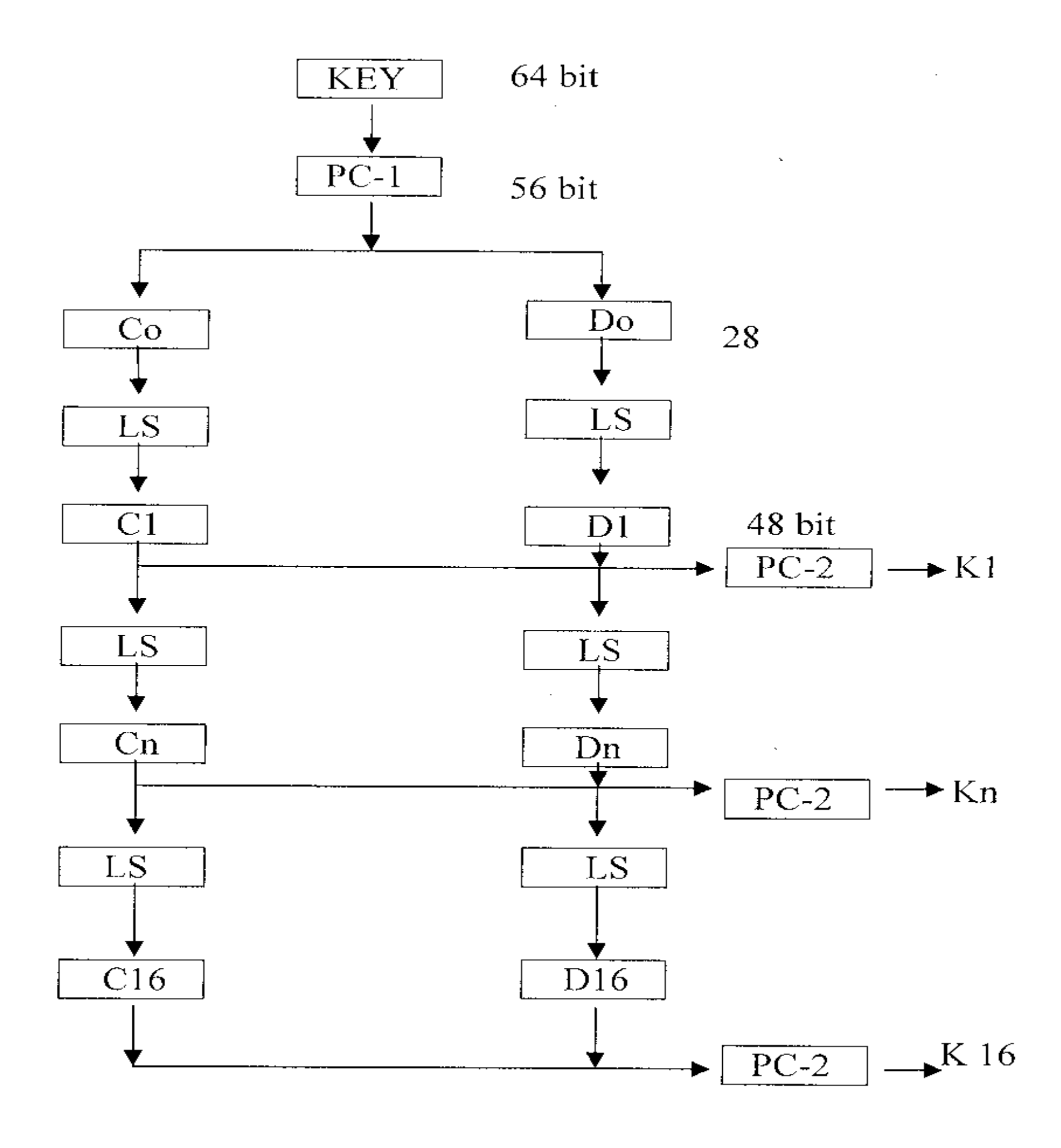
*** يتكون المفتاح الرئيسي K من64 بت ، تكون البتات 64,56,48,40,32,24,16,8 هي البتات الكاشفة للخطأ وبعد إسقاطها نحصل على 56 بت يجري لها ترتيبا أوليا PC-1 جدول (3) ، وهذا الجدول يقسم الإخراج إلى جزأين جزء أيسرC_0 وجزء أيمن D_0 وكل جزء يحتوي على 28 بت.**

*** بعد تحديد D_0, C_0 علينا إن نعرف كيف نجد D_n, C_n حيث إن إيجاد D_n, C_n من كتلتي البيانات D_{n-1}, C_{n-1} (n =1,2. ...16) وذلك باستخدام جدول التزحيفات لليسار LS جدول (4) وبالاعتماد على رقم الدورة.**

مثلا: إن كتلة البيانات D_3,C_3 نجدها بتزحيف كتلة البيانات D_2,C_2 مرتين الى اليسار. وكتلة البيانات D_{16},C_{16} نجدها بتزحيف كتلة البيانات D_{15},C_{15} مرة واحدة إلى اليسار على التوالي.

وعملية تزحيف موقع واحد إلى اليسار لكتلة 28 بت تعني إن تسلسل البتات كالأتي 1 ,28...2,3 بدلا من 1, 28...2 إي ما معناه محتويات الموقع 1 تنقل الى الموقع 28 ومحتويات الموقع 28 تنقل الى الموقع 27 وهكذا.

*** بعدها يتم إدخال الكتلة ذا ألـ56 بت إلى جدول اختيار الترتيب PC-2 (جدول 5) حيث إن البتات تحول وتقلص الى كتلة ذا 48 بت لتعطي المفتاح K. حيث إن رقم14 يمثل البت الأول من المفتاح والبت 17 يمثل البت الثاني وهكذا.**



شكل (5) عملية توليد المفتاح

8- خوارزمية DES:

إن DES مكون من خوارزمية تشفير للبيانات (DEA) وهذة الخوارزمية يتم إتباعها خطوة بعد خطوة والتي تتضمن عملية التشفير وفك شفرة المقطع المتكون من 64 بت كإدخال تحت تأثير مفتاح طوله 64 بت ليعطي إخراج بطول 64 بت ولكن بصورة مشفرة ، إن خطوات فك الشفرة يتم إتباعها بنفس خطوات التشفير ولكن بترتيب معكوس .إن الكتلة المطلوب تشفيرها يتم ترتيبها ترتيبا أوليا (IP) بالاعتماد على جدول(1) حيث إن البت 58 يكون البت الأول يتبعه البت 50 وهكذا ، ثم تجري عليها عمليات معقدة باستخدام مفتاح التشفير الذي يولد عشوائيا ثم يعكس الترتيب الأولي باستخدام جدول الترتيب الأولي المعكوس (IP^{-1} جدول(8)) ليعطي النص المشفر للكتلة المستخدمة.

وسوف نلخص هذه الخوارزمية بالخطوات التالية :-

1- **يدخل المستخدم 64 بت كمفتاح.**
2- **حساب المفاتيح الفرعية.**
3- **ينفذ الترتيب الاولي للمفتاح ذات 64 بت ويقلل الى 56 بت حسب جدولPC-1 رقم (3) .**
4- **نفصل المفتاح المرتب الى قسمين الاول 28 ويسمى C[0] والثاني 28 ويسمى D[0] .**
5- **نحسب المفتاح الفرعي ونبدا من I=1**
6- **نمثل واحد او اثنين من الدوائر المزاحة الى اليسار C[n-1] , D[n-1] لنحصل على C[n] , D[n] .**

//* عدد الدورات المزاحة سوف نبدا بها من 16......1,2 وتعتمد الازاحة على رقم الدورة وحسب جدول الإزاحات رقم (4) .

7- **اجراء الترتيب C[n] , D[n] لينتج لدينا مفتاح طوله 48 بت وحسب جدول PC-2 رقم (5) .**
8- **الخطوات من (1...7) تستمر حتى تكمل 16 دورة للمفتاح فقط .**
9- **معالجة البلوك المكون من 64 بت**
10- **تنفيذ عملية الترتيب الاولي على البلوك المحدد وحسب جدولIP رقم (1) .**
11- **نفصل البلوك الى قسمين الاول من 32 بت ويسمى L[0] والثاني من 32 بت ويسمى R[0] .**
12- **نطبق المفاتيح الفرعية على بلوك البيانات ونبدأ من I=1 .**
13- **نوسع 32 بت R[i-1] الى 48 بت وفقاً لدالة اختيار البت وحسب جدول اختيار البت-E رقم(2) .**
14- **نجري عملية XOR R[i-1] مع K[i] .**
15- **ندخل نتيجة الخطوة (14 السابقة) الى صناديق التعويض (جدول 6) والتي هي عبارة عن 8 صناديق تبدأ من 1..6 في B[1] ومن 7..12 في B[2] وهكذا .**
16- **نجد القيم التعويضية في صناديق التعويض والتي تبدأ من j =1 وكل صندوق يجب ان يمتلك 6 مدخلات و4 مخرجات (كما في الشكل (4)) .**
17- **ناخذ السطر الاول مع السادس ليمثلان رقم الصف والسطر الثاني مع الخامس ليمثلان رقم العمود .**

18- نقاطع قيمة الصف B[j] مع قيمة العمود S[i] في صناديق التعويض لنستخرج القيمة وحسب جدول (6) .

19- هذة الدورة من الخطوة (17) تعاد ليتم تبديل البلوكات 8 كلها .

20- ان الاخراج أي الناتج من صناديق S سوف يرتب لكل من B[1]…B[8] وحسب جدول الترتيب Pرقم(7) .

21- نجري عملية XOR للقيم الناتجة مع L[n-1] وكما في المعادلة الاتية :

R[n]=L[n-1] XOR p((S[1]B[1] …S[8] (B[8]))

//* عندما B[j] هي 6 لكل بلوك وتكون بالشكل الاتي E(R[n-1] XOR K[n]

ولغموض هذه المعادلة فأننا سنكتبها بالشكل الاتي :

R[n] = L[n-1] XOR F(R[i-1] , K[n])

22 - هذه الدوارة تستمر حتى تطبق 16 دورة

23 - واخيراً نرتب البلوك R[16] , L[16] وحسب جدولIP^{-1} رقم (8).

9- مثال/ عن كيفية استخدام DES في عملية التشفير:

لنفرض انه m ثمثل النص الصريح للرسالة والتي يكون حجمها 64 بت وتحتوي على ما يلي:

M=0123456789ABCDEF

ولنفرض ايضاً ان المفتاح k المستخدم للتشفير والذي ايضاً حجمه 64 بت هو الاتي :

K=133457799BBCDFF1

ملاحظة // قبل البدء بعملية التسفير سوف يتم تحويل النص المراد تشفيره ومفتاح التشفير الى صيغة التمثيل الثنائي وكما يلي :

M=0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111

K=00010011 00110100 01010111 01111001 10011011 10111100 11011111 11110001

ولكي تتم عملية التشفير فأننا سوف نتبع الخطوات الأتية :-

الخطوة الاولى :- حساب جدولة المفاتيح الفرعية لكل دورة بحيث يكون عدد المفاتيح الفرعية 16 مفتاح موزعة على عدد الدورات . وتكون هذه العملية بعدة خطوات وكما يلي :

1- ان المفتاح الذي حجمه 64 بت سوف يدخل الى جدول اختيار الترتيب PC-1 (جدول 3) ليقلص إلى 56 بت، بحيث ان المدخل الاول في الجدول هو 57 هو بت المفتاح العام سوف يصبح اول بت في جدول الترتيب والبت 49 هو بت المفتاح العام سوف يصبح البت الثاني لجدول الترتيب ، بينما البت 4 فأنه يمثل اخر بت في الجدول.

وعند تطبيق هذه العملية على كتلة المفتاح العام فأننا سوف نحصل على مفتاح ذات طول 56 بت بدلاً من 64 بت وكما يلي:

K=00010011 00110100 01010111 01111001 10011011 10111100 11011111 11110001
64 Bit
K+ = 1111000 0110011 0010101 0101111 0101010 1011001 1001111 0001111
56 Bit

2- سوف نقسم 56بت الى جزئين حيث كل جزء يتكون من 28 بت ويكون الجزء الايسر C_0 والجزء الأيمن D_0 .

C_0 = 1111000 0110011 0010101 0101111
D_0 = 0101010 1011001 1001111 0001111

ان C_0 وD_0 سوف نعرفهما لـ16 بلوك بحيث ان C_n و D_n تبدء من 16=>n=>1 وكل جزء من البلوكات سوف يحصل على نتيجة C_n و D_n من البلوك الذي قبلهُ اي C_{n-1} و D_{n-1} ويبدأ من 1...16 ويعتمد هذا على رقم الدورة(جدول 4)، فالدورة الاولى تكون عدد الازاحات موقع واحد لكل من C_0 وD_0 إي أن البت 2 يصبح 1 والبت 3 يصبح 2 وهكذا ، وهذا يقودنا إلى انه C_3 وD_3 نحصل عليها من C_2 وD_2 وتستمر الى نحصل على C_{16} وD_{16} من C15 وD15.

C_0 = 1111000011001100101010101111
D_0 = 0101010101100110011110001111

C_1 = 1110000110011001010101011111
D_1 = 1010101011001100111100011110

C_2 = 1100001100110010101010111111
D_2 = 0101010110011001111000111101

C_3 = 0000110011001010101011111111
D_3 = 0101011001100111100011110101

C_4 = 0011001100101010101111111100
D_4 = 0101100110011110001111010101

C_5 = 1100110010101010111111110000
D_5 = 0110011001111000111101010101

C_6 = 0011001010101011111111000011
D_6 = 1001100111100011110101010101

C_7 = 1100101010101111111100001100
D_7 = 0110011110001111010101010110

C_8 = 0010101010111111110000110011
D_8 = 1001111000111101010101011001

C_9 = 0101010101111111100001100110
D_9 = 0011110001111010101010110011

C_{10} = 0101010111111110000110011001
D_{10} = 1111000111101010101011001100

C_{11} = 0101011111111000011001100101
D_{11} = 1100011110101010101100110011

C_{12} = 0101111111100001100110010101
D_{12} = 0001111010101010110011001111

C_{13} = 0111111110000110011001010101
D_{13} = 0111101010101011001100111100

C_{14} = 1111111000011001100101010101
D_{14} = 1110101010101100110011110001

C_{15} = 1111100001100110010101010111
D_{15} = 1010101010110011001111000111

C_{16} = 1111000011001100101010101111
D_{16} = 0101010101100110011110001111

3- أن الكتلة ذات 56 بت الناتجة من C_0 وD_0 سوف تدخل الى جدول اختيار البت PC-2، وذلك لكي تقلص البتات الى 48 بت بحيث ان البت الأول لـ K_n يقابل البت 14 لـC_n وD_n والبت الثاني يقابل 17 وهكذا، وبتطبيق هذه الخطوة فإننا سوف نحصل على 16 مفتاح فرعي وكالاتي:

C_1D_1 = 1110000 1100110 0101010 1011111 1010101 0110011 0011110 0011110
K1 = 000110 110000 001011 101111 111111 000111 000001 110010
K_2 = 011110 011010 111011 011001 110110 111100 100111 100101
K_3 = 010101 011111 110010 001010 010000 101100 111110 011001
K_4 = 011100 101010 110111 010110 110110 110011 010100 011101
K_5 = 011111 001110 110000 000111 111010 110101 001110 101000
K_6 = 011000 111010 010100 111110 010100 000111 101100 101111
K_7 = 111011 001000 010010 110111 111101 100001 100010 111100
K_8 = 111101 111000 101000 111010 110000 010011 101111 111011
K_9 = 111000 001101 101111 101011 111011 011110 011110 000001
K_{10} = 101100 011111 001101 000111 101110 100100 011001 001111
K_{11} = 001000 010101 111111 010011 110111 101101 001110 000110
K_{12} = 011101 010111 000111 110101 100101 000110 011111 101001
K_{13} = 100101 111100 010111 010001 111110 101011 101001 000001
K_{14} = 010111 110100 001110 110111 111100 101110 011100 111010
K_{15} = 101111 111001 000110 001101 001111 010011 111100 001010
K_{16} = 110010 110011 110110 001011 000011 100001 011111 110101

ملاحظة // انتهت عملية توليد المفاتيح الفرعية

الخطوة الثانية:- **عملية تشفير بلوك البيانات ويتم بعدة خطوات :**

1- ادخال بلوك البيانات الذي حجمه 64 بت الى جدول الترتيب الاوليIP(جدول 1)، بحيث يحرك البت في الموقع 58 الى الموقع1 والبت في الموقع 50 الى الموقع 2 وهكذا، وعند تطبيق هذه العملية على الرسالة m فأننا نحصل على النتيجة الآتية:

M = 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111
IP = 1100 1100 0000 0000 1100 1100 1111 1111 1111 0000 1010 1010 1111 0000 1010 1010

2- نقسم البلوك الذي حصلنا عليه من IP الى جزئين ايسر L_0 و أيمن R_0 بحيث كل جزء يتكون من 32 بت . وتستمر هذه العملية الى ان نحصل على L_{16} وR_{16} .

L_0 = 1100 1100 0000 0000 1100 1100 1111 1111
R_0 = 1111 0000 1010 1010 1111 0000 1010 1010

3- نوسع البلوك R_{n-1} لكي يصبح 48 بدلاً من 32 وذلك بواسطة جدول اختيار البت E . (لاحظ جدول 2) ، وتكون عملية حساب $E(R_0)$ من R_0 كالتالي :

R_0 = 1111 0000 1010 1010 1111 0000 1010 1010
$E(R_0)$ = 011110 100001 010101 010101 011110 100001 010101 010101

4- نحسب دالة F وكيفية عملها (راجع صفحة 10) من المعادلات الاتية :

$L_n = R_{n-1}$ n=1…………16
$R_n = L_{n-1} + f(R_{n-1},K_n)$ **وعند تطبيق المعادلتين فاننا نحصل على النتيجة آلاتية :**

K_1 = 000110 110000 001011 101111 111111 000111 000001 110010
$L_1 = R_0$ = 1111 0000 1010 1010 1111 0000 1010 1010
$R_1 = L_0 + f(R_0,K_1)$

5- نجري عملية XOR بين K_n و$E(R_{n-1})$ من خلال المعادلة الاتية وكما يلي :

$K_n \oplus E(R_{n-1})$

K_1 = 000110 110000 001011 101111 111111 000111 000001 110010
$E(R_0)$ = 011110 100001 010101 010101 011110 100001 010101 010101
$K_1 \oplus E(R_0)$ = 011000 010001 011110 111010 100001 100110 010100 100111

6- أن ناتج الخطوة السابقة (خطوة 5) سوف يقسم الى 8 مجاميع وذلك حسب صناديق S بحيث ياخذ كل صندوق 6 بت كادخال ويعطي 4 بت كاخراج وبذلك نحصل على 32 بت ، وكما يلي :
$K_n + E(R_{n-1}) = B_1B_2B_3B_4B_5B_6B_7B_8$,
$S_1(B_1)S_2(B_2)S_3(B_3)S_4(B_4)S_5(B_5)S_6(B_6)S_7(B_7)S_8(B_8)$ **كلB تأخذ 6 بت**

وكنتيجة عملية فاننا نحصل على ما يلي :

$K_1 + E(R_0)$ = 011000 010001 011110 111010 100001 100110 010100 100111.
$S_1(B_1)S_2(B_2)S_3(B_3)S_4(B_4)S_5(B_5)S_6(B_6)S_7(B_7)S_8(B_8)$ = 0101 1100 1000 0010 1011 0101 1001 0111

7- يتم ادخال 32 بت الناتجة من صناديق S الى جدول الترتيب P (جدول 7) ليتم حساب قيم F النهائية وكالاتي:

$f = P(S_1(B_1)S_2(B_2)...S_8(B_8))$
$S_1(B_1)S_2(B_2)S_3(B_3)S_4(B_4)S_5(B_5)S_6(B_6)S_7(B_7)S_8(B_8)$ = 0101 1100 1000 0010 1011 0101 1001 0111
f = 0010 0011 0100 1010 1010 1001 1011 1011

8- إن النتيجة التي حصلنا عليها تمثل أول دالة تشفير من حسابات ألـ 16 دالة f(R,K) .
وألان ألـ 32 بت (الاقصى اليسار L_0) و f(R,K) تجمعان (جمع للأساس 2) لتعطي كتلة جديدة ذا 32 بت هي R_1 وحسب المعادلة التالية :

$R_1 = L_0 + f(R_0 , K_1)$ = 1100 1100 0000 0000 1100 1100 1111 1111
+ 0010 0011 0100 1010 1010 1001 1011 1011
= 1110 1111 0100 1010 0110 0101 0100 0100

وفي الدورة التالية يتم حساب $L_2=R_1$ وبذلك نحصل على $R_2 =L_1 + f(R_1, K_2)$ ونستمر الى ان نحصل على R_{16} ,L_{16} ومن ثم نعكس ترتيب البلوكين ليصبح R_{16} ,L_{16} .

9- نقوم بأجراء الترتيب الأولي المعكوس IP^{-1} (جدول 8) ، حيث ان البت 40 يمثل البت الاول والبت 8 يمثل الثاني وهكذا الى ان نصل الى البت 25 والذي يمثل البت الاخير ، وعند تطبيق هذه الخطوة على البلوك الاخير R_{16},L_{16} فاننا نحصل على ما يلي :

L_{16} = 0100 0011 0100 0010 0011 0010 0011 0100
R_{16} = 0000 1010 0100 1100 1101 1001 1001 0101
$R_{16}L_{16}$ = 00001010 01001100 11011001 10010101 01000011 01000010 00110010 00110100
IP^{-1} = 10000101 11101000 00010011 01010100 00001111 00001010 10110100 00000101

واخيراً نحصل على النص المشفر للرسالة M ويكون كالآتي :

M = 0123456789ABCDEF   النص الصريح
C = 85E813540F0AB405   النص المشفر

* ولكي تتم عملية فك التشفير فانه يتم اتباع نفس الخطوات السابقة ولكن بصورة معكوسة .

10- تقييم نظام DES:

إن وضع نظام قياسي لحماية المعلومات في الحاسبة الالكترونية يعتبر بحد ذاته ثورة في علم التشفير وتعتبر تحدي كبير لمحطمي الشفرة وذلك كون الخوارزمية منشورة وبمتناول اليد والشئ الوحيد المجهول هو المفتاح.

ومما يؤاخذ عليه نظام DES هو الاستخدام الواسع له حيث إن محطم الشفرة يعلم بأنه بكسر الشفرة فانه سيجني فائدة كبيرة وذلك بسرقة معلومات لأكثر من مستخدم مما يجعل عملية مهاجمته وكسره مشجعة واقتصادية ، في حين إن عملية مهاجمة نظام تشفير يستخدم من قبل مستفيد واحد تكون مكلفة جدا .

ويعاني نظام التشفير القياسي من مشكلة رئيسية متفق عليها هي مشكلة توزيع وإدارة المفاتيح. حيث أن الخوارزمية تحتوي على مفتاح واحد عام تتولد منه المفاتيح الفرعية والتي بدورها سوف تتوزع على 16 دورة

، لذلك فانه بمجرد معرفة ذلك المفتاح من قبل المهاجم فان الخوارزمية سوف تنكسر وتصبح عديمة الفائدة وتنكشف المعلومات المحمية بداخلها، ولهذا السبب تم استحداث خوارزمية تشبه في عملها ومضمونها خوارزمية DES ولكنها تختلف في عدد المفاتيح المستخدمة حيث أنها تستخدم ثلاثة مفاتيح مختلفة ومستقلة في عملية التشفير بدلاً من المفتاح الواحد ولهذا السبب سميت بخوارزمية التشفير القياسية المضاعفة Triple Data Encryption Standard (TDES) .

11- خوارزمية TDES:

أن خوارزمية تشفير البيانات القياسية المضاعفة Triple Data Encryption Standard (TDES). تعتبر تطوير لخوارزمية DES حيث أنها تعمل بنفس الطريقة التي تعمل بها DES من حيث عدد الدورات والعمليات الحسابية المعقدة، ولكنها تختلف فقط في عدد المفاتيح المستخدمة في التشفير، فالخوارزمية السابقة كانت تستخدم مفتاح واحد في عملية التشفير وهذا أدى إلى ظهور ضعف في خوارزمية DES بسبب أمكانية كسر أو اكتشاف مفتاح التشفير، ولهذا السبب تم تطوير الخوارزمية لتعالج تلك المشكلة وذلك من خلال استخدام ثلاثة مفاتيح مختلفة يعمل كل واحد منهما بشكل مستقل عن الأخر.

ويمكن أن نلخص خطوات خوارزمية TDES كما يلي:

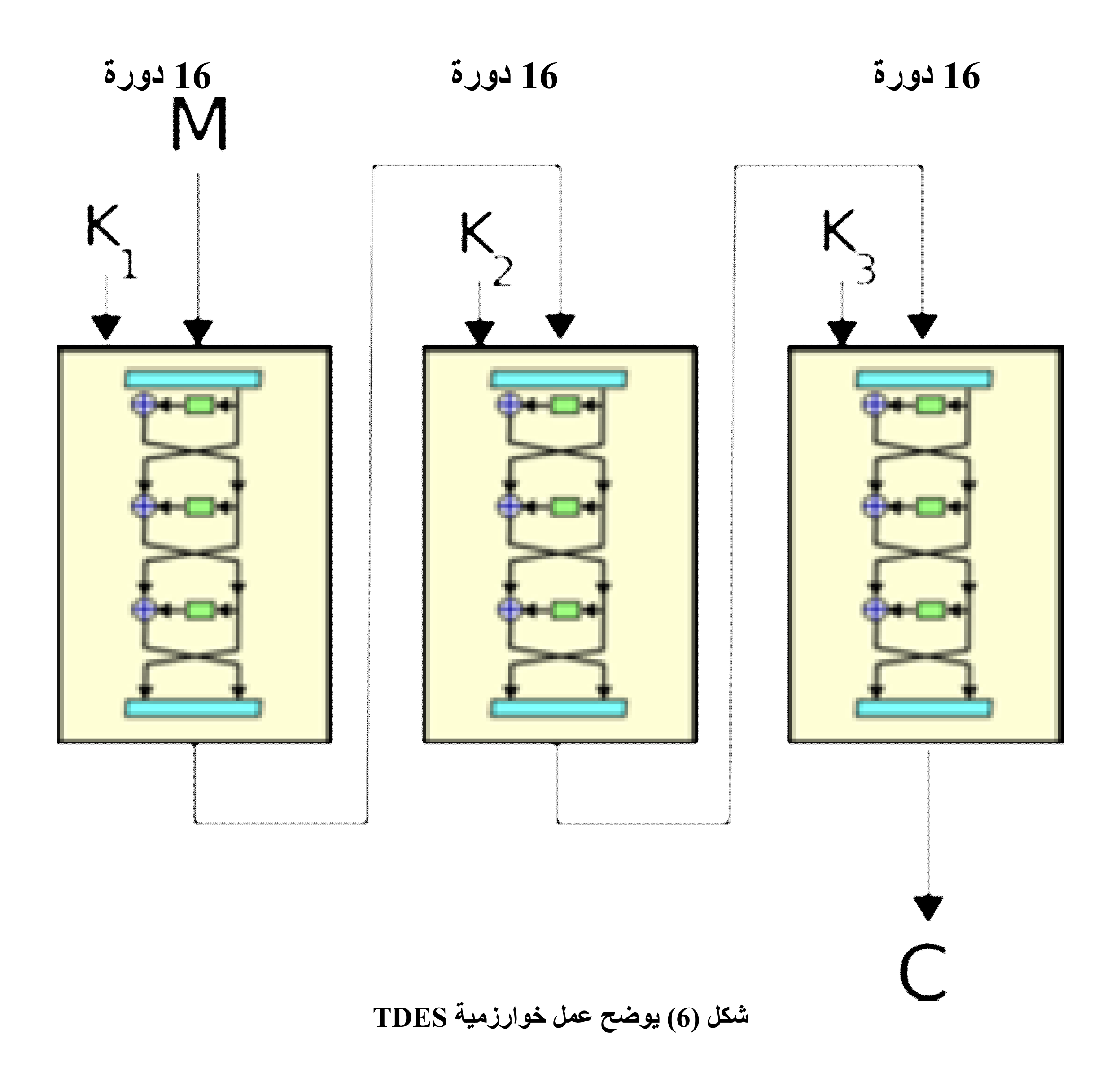
1- إتباع نفس خطوات خوارزمية DES، حيث تكون تلك العملية مع المفتاح الأول K_1، وتكون النتيجة النهائية لهذه الخطوة نص مشفر ولكن بصورة غير كاملة، أي انه بعد المرور بـ16 دورة فان النص الناتج يكون غير مشفر بصورة نهائية.

2- اخذ النتيجة من الخطوة الأولى وإعادة نفس الخطوات السابقة ولكن هذه المرة مع المفتاح الثاني K_2، وأيضاً تكون النتيجة نص مشفر ولكنه ليس النص النهائي المشفر.

3- نأخذ النص الناتج من الخطوة السابقة وأيضاً يتم إعادة خطوات الخوارزمية ولكن هذه المرة مع المفتاح الثالث K_3، وتكون هذه هي الخطوة النهائية في عملية التشفير بواسطة خوارزمية TDES. وينتج بعدها نص مشفر بصورة نهائية.و(الشكل(6)) يوضح عملية التشفير الخاصة بخوارزمية TDES .

TDES Encryption Operation:

DES EK1 , DES EK2 , DES EK3



شكل (6) يوضح عمل خوارزمية TDES

ولكي نجري عملية فك التشفير فانه يتم أتباع نفس الخوارزمية السابقة، ولكن بشكل معكوس عن عملية التشفير

TDES Decryption Operation:

DES DK3 , DES DK2 , DES DK1

12- نتائج المقارنة:

بعد وصف كلاً من خوارزميتي DES و TDES فانه يتبين لنا أن الخوارزميتين متشابهتان من حيث هياكل البيانات المستخدمة و الخطوات المتبعة في عمليتي التشفير وفك التشفير، ولكنهما مختلفتان في عدد المفاتيح المستخدمة للتشفير، وهذا الاختلاف في عدد المفاتيح أدى إلى ظهور عدد من الاختلافات في عمل كلاً من خوارزميةDES و TDES ولكي نجر ي عملية مقارنة دقيقة فأننا سوف نأخذ عدد من النقاط المهمة والتي تعتبر مهمة في عمل الخوارزميتين وهذه النقاط هي وقت كسر الخوارزمية و سرعة التنفيذ والأمنية وعدد مفاتيح التشفير وحجم الذاكرة التي تستهلكه كل من الخوارزميتين وأخيرا عدد الدورات التي تمر بها الخوارزميتين ولقد اعتمدنا في هذه المقارنة على برنامج مصمم بلغة الفيجويل بيسك لكلا الخوارزميتين، وسوف نقوم بشرح كل نقطة من هذه النقاط بالتفصيل وكما يلي:

<u>1- وقت كسر الخوارزمية:</u> أن خوارزمية DES تكون قابلية كسرها اكبر بمرتين من خوارزمية TDES، وذلك بسبب المفتاح الواحد المستخدم في عملية التشفير، حيث إن المهاجم الذي يحاول كسر الخوارزمية وبمجرد أن يكتشف المفتاح المستخدم في التشفير فان الخوارزمية سوف تنكسر وتصبح عديمة الفائدة.
بينما خوارزمية TDES تكون صعبة الكسر وذلك لأن المهاجم أذا اكتشف المفتاح الأول K_1 فانه يبقى لديه مفتاحان K_2 وK_3 لكي يحصل على النص الصريح أو المعلومات المشفرة، وذلك لأنه بعد معرفة المفتاح الأول فان النتيجة تكون أيضا نص مشفر وغير مفهوم، وهذا يؤدي إلى استغراق وقت أطول لكسر الخوارزمية ومعرفة المفتاحان المتبقيان من قبل المهاجم .

<u>2- سرعة التنفيذ:</u> إن خوارزمية DES تكون عملية تنفيذها سريعة مقارنةً مع خوارزمية TDES وذلك لأنه عملية التشفير تحدث مع مفتاح واحد فقط لـ16 دورة وبعد هذه العملية ينتج لدينا نص مشفر نهائي.
بينما خوارزمية TDES تكون أبطأ لأنها تعادل خوارزمية DES بمرتين من حيث العمليات الحسابية، ففي الـ16 دورة الأولى تجري عمليات حسابية معقدة جداً مع كتلة النص الصريح والمفتاح الأول K_1 ، وتحدث نفس العمليات في 16 دورة ثانية ولكن هذه المرة مع المفتاح الثاني K_2 ، وهكذا إلى إن نصل إلى المفتاح الثالث K_3 ، ونلاحظ في نهاية الأمر إن خوارزمية TDES هي تكرار لخوارزمية DES لمرتين مع مفتاحين مختلفان ومستقلان في العمل وبالتالي فإنها تكون أبطأ من DES بسبب ازدياد العمليات الحسابية المعقدة.

3- الأمنية: مؤكد وبدون شك إن خوارزمية TDES تكون ذات أمنية أعلى واكبر من خوارزمية DES وذلك كما وضحنا بأن المهاجم في خوارزمية TDES يجب أن يكتشف أو يكسر ثلاثة مفاتيح مختلفة لكي يحصل على النص الصريح.

بينما في خوارزمية DES تكون أمكانية كسر مفتاح التشفير كبيرة، فالمهاجم إذا تمكن من كسر مفتاح التشفير أصبح بإمكانه كسر الخوارزمية والاطلاع على المعلومات المشفرة، ولا ننسى إن سبب تطوير خوارزمية DES إلى TDES هو لزيادة أمنية الخوارزمية ومعالجة مشكلة ضعف توزيع وإدارة المفاتيح الفرعية.

4- عدد مفاتيح التشفير: أن خوارزمية DES كما وضحنا سابقاً تستخدم مفتاح واحد في عملية التشفير وهذا هو سبب ضعفها كخوارزمية تشفير تكاد تكون عالمية ولهذا فأن خوارزمية TDES عملت على معالجة تلك المشكلة من خلال استخدام ثلاثة مفاتيح مختلفة ومستقلة في عملها في عملية التشفير، وبهذا زادت قوة الخوارزمية وأمنيتها.

5- حجم الذاكرة: تأخذ خوارزمية TDES ذاكرة اكبر من DES وذلك بسبب كبر العمليات الحسابية التي تحدث داخلها ، حيث تجري مع كل من المفاتيح الثلاثة المستخدمة في التشفير عمليات حسابية معقدة مثل التزحيفات وحساب دالة F وعمليات صناديق التعويض S-BOX وغيرها.

6- عدد الدورات: أن خوارزمية TDES تكون عدد دوراتها اكبر من DES التي عدد دوراتها 16 دورة فقط أثناء عملية التشفير الكلية ، فهي كما شرحنا تستخدم ثلاثة مفاتيح مستقلة وكل مفتاح يمر بـ16 مستقلة عن المفتاح الأخر ، أي أن عدد الدورات يكون 3 x 16 وهذا بدوره يعطينا 48 دورة للخوارزمية الكلية.

والجدول التالي يوضح صفات المقارنة بين خوارزميتي DES و TDES بشكل ملخص:

الخصائص	*خوارزمية DES*	*خوارزمية TDES*
وقت كسر الخوارزمية	قصير	تستغرق وقتا كبيراً
سرعة التنفيذ	سريعة التنفيذ	اقل سرعة
الأمنية	ذات أمنية قليلة	ذات أمنية عالية

عدد مفاتيح التشفير	تستخدم مفتاحاً واحد	تستخدم ثلاثة مفاتيح
حجم الذاكرة	تستهلك ذاكرة قليلة	تستهلك ذاكرة اكبر
عدد الدورات	16 دورة	48 دورة

13- الاستنتاجات:

بعد الدراسة التي قمنا بها للمقارنة بين خوارزميتي التشفير DES وTDES تم استنتاج مايلي :

1- أن خوارزمية التشفيرDES هي خوارزمية قوية وفعالة حيث عملت لسنوات طويلة في أكثر من مجال ، ولكنها في الآونة الأخيرة عانت من مشاكل عديدة منها ضعف المفتاح المستخدم في التشفير وقابلية كسره من قبل المهاجم وهذا بدوره أدى إلى ضعف الخوارزمية وقلة استخدامها.

2- أن ضعف خوارزمية DES أدى إلى استحداث خوارزمية جديدة مشتقة من نفس هيكلة خوارزمية DES ولكنها تختلف عنها في عدد المفاتيح المستخدمة في التشفير حيث أنها تستخدم ثلاثة مفاتيح مختلفة، ولهذا سميت بخوارزمية التشفير المضاعفة TDES، وهذا ساعد بدوره إلى زيادة قوة الخوارزمية وبالتالي زيادة أمنيتها وهذا هو المطلوب في عمل أي خوارزمية تشفير.

3- أن خوارزمية TDES تشبه خوارزمية DES من حيث هياكل البيانات المستخدمة والعمليات الحسابية المعقدة التي تحدث بين كتلة النص الصريح ومفتاح التشفير في الـ16 دورة ، ولكن هناك عدة نقاط تختلف بها الخوارزميتين وهي:

أ- خوارزمية TDES تستخدم ثلاثة مفاتيح للتشفير مستقلة في عملها بدلاً من المفتاح الواحد كما يحدث في خوارزمية DES.

ب- خوارزمية TDES تستغرق وقتاً كبير لكسرها بسب احتوائها على ثلاثة مفاتيح مختلفة وهذا يزيدها صعوبة اكبر بالنسبة للمهاجم .

ج- خوارزمية TDES بطيئة التنفيذ مقارنة بخوارزمية DES ، لأنها تجري عملية التشفير لثلاث مرات مع ثلاثة مفاتيح وفي كل مرة تزداد العمليات الحسابية تعقيداً وهذا يؤدي إلى بطئها في التنفيذ.

د- أن خوارزمية TDES هي ذات أمنية أعلى من خوارزمية DES وذلك بسبب مضاعفة عدد مفاتيح التشفير والذي أدى إلى زيادة وصعوبة العمليات الحسابية المعقدة التي تحدث داخل الخوارزمية.

ه - خوارزمية TDES تستهلك ذاكرة اكبر حجماً من التي تستهلكها خوارزمية DES.

و- وأخيرا فان خوارزمية TDES تكون عدد دوراتها كبير حيث تكون 48 دورة وذلك عكس خوارزمية DES التي تكون عدد دوراتها 16دورة فقط.

المصادر:

1- J. Orlin Grabbe, "the DES Algorithm Illustrated"

2- Raymond G. Kammer, Director, "DATA ENCRYPTION STANDARD ، (1999) (DES)"

3- د. وسيم الحمداني ، " أنظمة التشفير" ، 1997.

ميمونة الحداد
shrm_4fra@yahoo.com